INTERSPEECH 2016
September 8–12, 2016, San Francisco, USA

A KL Divergence and DNN-based Approach to Voice Conversion without Parallel Training Sentences

Feng-Long Xie[1,2*], *Frank K. Soong*[2], *Haifeng Li*[1]

[1]Harbin Institute of Technology, Harbin, China
[2]Microsoft Research Asia, Beijing, China
{v-fxie,frankkps}@microsoft.com, lihaifeng@hit.edu.cn

Abstract

We extend our recently proposed approach to cross-lingual TTS training to voice conversion, without using parallel training sentences. It employs Speaker Independent, Deep Neural Net (SI-DNN) ASR to equalize the difference between source and target speakers and Kullback-Leibler Divergence (KLD) to convert spectral parameters probabilistically in the phonetic space via ASR senone posterior probabilities of the two speakers. With or without knowing the transcriptions of the target speaker's training speech, the approach can be either supervised or unsupervised. In a supervised mode, where adequate training data of the target speaker with transcriptions is used to train a GMM-HMM TTS of the target speaker, each frame of the source speakers input data is mapped to the closest senone in thus trained TTS. The mapping is done via the posterior probabilities computed by SI-DNN ASR and the minimum KLD matching. In a unsupervised mode, all training data of the target speaker is first grouped into phonetic clusters where KLD is used as the sole distortion measure. Once the phonetic clusters are trained, each frame of the source speakers input is then mapped to the mean of the closest phonetic cluster. The final converted speech is generated with the max probability trajectory generation algorithm. Both objective and subjective evaluations show the proposed approach can achieve higher speaker similarity and better spectral distortions, when comparing with the baseline system based upon our sequential error minimization trained DNN algorithm.

Index Terms: voice conversion, Kullback-Leibler divergence, deep neural networks

1. Introduction

Voice Conversion is to convert the speech from a source speaker to a target speaker without changing the word content. Conventional voice conversion techniques[1, 2, 3, 4, 5, 6, 7, 8, 9, 20, 21] always need parallel data which can be automatically aligned by dynamic time warping. A mapping function is then trained to convert the speech from source speaker to the target speaker. Among these approaches, the joint density Gaussian mixture model(JD-GMM) based mapping method [2] and neural network (NN) based mapping method [5] are widely used. Although JD-GMM can effectively convert source speech to target speech with a decent quality, there still exists some over smoothing problem due to the statistical averaging in training the mean and covariance of Gaussian components. NN based approach directly train the conditional probability which converts source speech to target speech. Besides, the conversion function of NN based approach is non-linear which might be able to simulate some non-linear function in speech production/perception. So NN has a potential to achieve better performance than GMM based approach [4]. Recently, exemplar-based sparse representation for voice conversion is studied[20, 21], the basic idea is to represent magnitude spectrum as a linear combination of a set of basis spectra, called the exemplars with non-negative matrix factorization, and try to convert the exemplars. However the precondition of having parallel data is inconvenient and for speaker pairs, we need to collect parallel recordings and train the corresponding mapping function which may involve expensive manual work. There are some approaches which do not require parallel data. In [13], acoustic clusters are constructed based on the acoustic features for source and target speaker respectively, and then a mapping between source and target acoustic clusters is established in terms of Euclidean distance. In [14] a speech recognizer is used to index each frame of source and target with state label. And then subsequences are extract from the set of target sequences to match the given source state index sequences. Thus parallel data of source and target speaker is constructed and the conventional linear transformation parameter training is applied. In [15] a unit selection based method is used to select the acoustically "nearest" target frame considering the continuity at the same time. However all three approaches can not achieve as good performance as GMM or NN based voice conversion which requires parallel data[15].

In this paper we propose a new voice conversion method for any unknown source speaker with or without his/her prerecorded speech. It is motivated by our new cross-lingual TTS work [16] which uses a SI-DNN to equalize speaker difference between different languages and Kullback-Leibler divergence to measure the phonetic distortion between two acoustic segments. We extend this KLD-DNN approach to voice conversion. A SI-DNN ASR is trained and the corresponding ASR senones space is used to represent the whole phonetic space independent of speaker. Speaker differences can be equalized with the SI-DNN at the senone or frame level in the ASR senone phonetic space. In a supervised mode when there is adequate training data of the target speaker with transcriptions. Each frame of source speaker is mapped to the TTS senones of the target speaker with minimum KLD calculated from the SI-DNN output posterior probabilities. In unsupervised mode when no transcriptions of the target speaker's speech are needed. Target speaker's phonetic clusters are constructed with KLD and each source speaker's acoustic frame is mapped to target speaker's phonetic cluster via KLD matching. Maximum probability trajectory speech generation algorithm is then applied to generate the converted speech trajectory.

*Work performed as an intern in the Speech Group, Microsoft Research Asia

http://dx.doi.org/10.21437/Interspeech.2016-116

2. KL Divergence And DNN Approach To Voice Conversion

2.1. Senone Mapping

A block diagram of senone mapping in KLD-DNN based voice conversion is shown in Fig. 1.

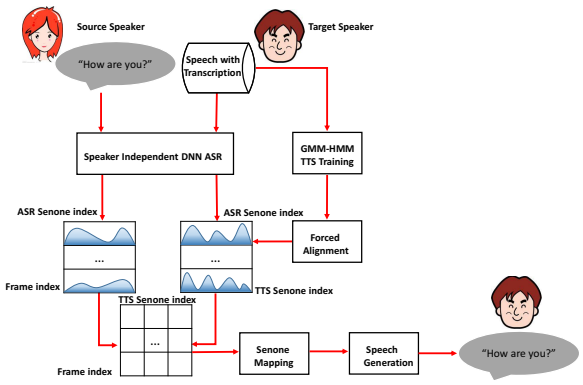

Figure 1: KLD-DNN: Senone Mapping

In training, only the target speaker's speech is needed and it is used to train a GMM-HMM based TTS of the target speaker. Forced align target speaker's speech with corresponding transcriptions, and we can get I buckets of training data, where each bucket is a group of acoustic frames with the same TTS senone label and I is the number of target speaker's TTS senones, or the clustered GMM states. Each TTS senone s_i has its corresponding acoustic mean vector μ_i and covariance matrix U_i. Then we process each bucket of target speaker's data via the SI-DNN to get the accumulated posteriors for all N English ASR senones ($s_1^{ASR}\ s_2^{ASR}\ ...\ s_N^{ASR}$), where N is the number of English ASR senones. The accumulated posteriors are then averaged by the number of frames in each bucket. For each TTS senone s_i, the ASR senone posterior distribution P_i is:

$$\begin{aligned} P_i &= [\ p(s_1^{ASR}|s_i)\ \ p(s_2^{ASR}|s_i)\ \ ...\ \ p(s_N^{ASR}|s_i)\ ], \\ p(s_n^{ASR}|s_i) &= \frac{\sum_{r=1}^{R} p(s_n^{ASR}|x_r)}{R},\ \ x_r \in bucket_i \end{aligned} \tag{1}$$

x_r is the frame belongs to $bucket_i$, R is the total number of frames in $bucket_i$. $i \in [1, 2, ..., I]$.

In conversion, for a source speaker, we process an utterance $X = [x_1, x_2, ..., x_T]$ with the SI-DNN, where T is the number of frames. And for each frame x_t, we obtain the ASR senone posterior distribution Q_t,

$$\begin{aligned} Q_t = [\ q(s_1^{ASR}|x_t)\ \ q(s_2^{ASR}|x_t)\ \ ...\ \ q(s_N^{ASR}|x_t)\ ], \\ t \in [1, 2, ..., T] \end{aligned} \tag{2}$$

We use symmetrised KL divergence [11] to measure phonetic distortion between each TTS senone s_i with distribution P_i and acoustic frame x_t with distribution Q_t in the probability space. Senone mapping is established with the minimum KLD selection criterion. For frame x_t of the source speaker, we find a corresponding TTS senone $s_{\mathscr{M}(t)}$ of the target speaker in the minimum KLD sense, where $\mathscr{M}(t)$ returns the corresponding index of target speaker's TTS senone given the source speaker's acoustic frame x_t.

$$\begin{aligned} \mathscr{M}(t) &= \underset{i}{\operatorname{argmin}}\ \boldsymbol{D}_{KL}(P_i, Q_t) \\ &= \underset{i}{\operatorname{argmin}} \sum_{n=1}^{N} (p(s_n^{ASR}|s_i) - q(s_n^{ASR}|x_t)) * \\ &\quad (\ln(p(s_n^{ASR}|s_i)) - \ln(q(s_n^{ASR}|x_t))), \\ &\quad i \in [1, 2, ..., I],\ \ t \in [1, 2, ..., T] \end{aligned} \tag{3}$$

In final conversion, a smooth acoustic parameter C is generated by the maximum probability parameter generation algorithm with delta constrains,

$$C = (W^T U^{-1} W)^{-1} W^T U^{-1} M \tag{4}$$

where W are the dynamic feature coefficient matrix [10], $M = [\mu_{\mathscr{M}(1)}^T, \mu_{\mathscr{M}(2)}^T, ..., \mu_{\mathscr{M}(T)}^T]^T$, $U^{-1} = diag[U_{\mathscr{M}(1)}^{-1}, U_{\mathscr{M}(2)}^{-1}, ..., U_{\mathscr{M}(T)}^{-1}]$

2.2. Phonetic Cluster Mapping

In many application scenarios, the amount of the target speaker speech is limited and transcriptions are not available. For this case, we switch to the unsupervised mode and perform a phonetic clusters based instead of TTS senone based mapping as described in the previous section.

2.2.1. Phonetic Clustering

The followings are the pseudo code of clustering algorithm. In

Algorithm 1 KL Divergence based Phonetic Clustering

Input:
L acoustic frames, $X = [x_1\ \ x_2\ \ ...\ \ x_L]$;
and their phonetic representations in ASR senone (posterior probabilities) $P = [P_1\ \ P_2\ \ ...\ \ P_L]$;
# of phonetic clusters: K;

Output:
Phonetic representations of c_k of each cluster s_k; Acoustic mean μ_k of each cluster s_k; Acoustic variance σ_k^2 of each cluster s_k

1: Initialization: Randomly pick K samples from L samples as K centroids; $t = 0$; $D^0 = \infty$
2: **repeat**
3: Calculate the KL Divergence $D_{KL}(P_i, c_k)$ $k \in [1, 2, ..., K]$ $i \in [1, 2, ..., L]$
4: Assign each sample P_i to the nearest cluster s_k with the minimum KL divergence. $s_k = \{P_i : D_{KL}(P_i, c_k) \leq D_{KL}(P_i, c_m), \forall m, 1 \leq m \leq K\}$
5: Recompute the phonetic representation c_k of each clusters_k $c_K \approx \frac{1}{2} \times (\frac{1}{|s_k|} \sum_{P_i \in s_k} P_i + \frac{^{|s_k|}\sqrt{\prod_{P_i \in s_k} P_i}}{\sum\ ^{|s_k|}\sqrt{\prod_{P_i \in s_k} P_i}})$
6: Recompute the total Distortion $D^{t+1} = \sum_{k=1}^{K} \sum_{P_i \in s_k} D_{KL}(P_i, c_k)$, $t = t + 1$
7: **until** $\frac{D^t - D^{t+1}}{D^t} < 1\ ‰$
8: Calculate acoustic statistics of each cluster $\mu_k = \frac{1}{|s_k|} \sum_{x_i \in s_k} x_i$ $\sigma_k^2 = \frac{1}{|s_k|} \sum_{x_i \in s_k} x_i^2 - \mu_k^2$
9: **return** phonetic representation c_k and acoustic statistics μ_k, σ_k^2 of each cluster

statistical parametric TTS training, a decision tree is constructed by clustering context-depdent, hidden Markov Model (HMM) states. Here we divide the acoustic space into clusters by unsupervised clustering. Given the target speaker's speech, we can extract the acoustic featuers $X = [x_1\ x_2\ \ldots\ x_L]$ and their phonetic representations in DNN output posterior probabilities. The unsupervised clustering is very similar to k-means clustering with symmetrised KLD as the distortion measure. The centroids are computed either as geometric mean or arithmetic mean to approximate the real centroid.

2.2.2. *Phonetic Cluster Mapping*

A block diagram of phonetic cluster mapping in KLD-DNN based voice conversion is shown in Fig.2. It's quite similar to the senone mapping. However, instead of TTS senones, we use the centroids of phonetic clusters to do KLD matching. The phonetic cluster is achieved by clustering the phonetic representations (DNN output posterior probabilities).

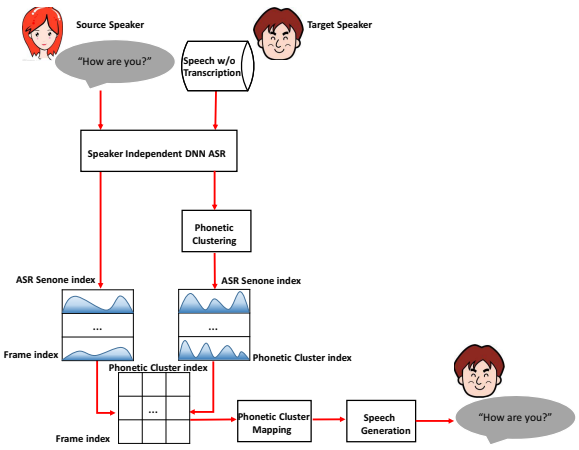

Figure 2: KLD-DNN: Phonetic Cluster Mapping

2.3. Speaker Dependent DNN

Given the source and target speaker's pre-recorded speech with transcriptions, we can further improve the KLD-DNN based voice conversion's performance by adapting the SI-DNN to speaker dependent DNN.

We adapt the SI-DNN to source speaker's DNN and target speaker's DNN respectively by simply finetuning the last layer of SI-DNN. The SD-DNN output posterior is more phonetically accurate to the specific source and target speaker. The speaker difference is equalized in the SD-DNN output ASR senone space. The succeeding mapping can be done in senone or phonetic cluster mapping as mentioned in previous two subsections.

2.4. Prosody Transformation

In this study we concentrate on preserving the speaker's spectral characteristics. The prosody is transformed in a global scale between the source and the target speakers . In conversion stage, a Gaussian normalized transformation[4] is used to transform the $F0$ of the source speaker to the $F0$ of the target speaker as follows:

$$\ln(F0_{Trans}) = \mu_t + \frac{\sigma_t}{\sigma_s}(\ln(F0_s) - \mu_s) \qquad (5)$$

where μ_s σ_s and μ_t σ_t are the means and standard deviations of the source and target speaker's F0, respectively.

3. Experiments

Experiments were carried out on the CMU ARCTIC database[12]. The ARCTIC corpus consists of four primary sets of recordings of 4 speakers (2 male BDL and RMS, 2 female CLB and SLT), plus 3 other accented sets of recordings (3 male, Canadian JMK, Scottish AWB and Indian KSP). In our experiments, we do voice conversion in 3 pairs:

1) SLT (US Female) to BDL (US Male)
2) SLT (US Female) to CLB (US Female)
3) RMS (US Male) to BDL (US Male)

3.1. Experimental Setup

1,000 English utterances ($\sim$ 1 hour) of a target speaker are used for training speaker dependent, English GMM-HMM based TTS. Speech is sampled at 16kHz, windowed by a 25ms windows, and shifted every 5ms. 40th-order Line Spectral Pair (LSP) coefficients [17] plus gain and the corresponding first and second order dynamic features, the fundamental frequency(F0) in log scale and its first and second order dynamic features are extracted. Multi-space probability Distribution (MSD) HMMs of 5-states, left to right, no-skip topology with diagonal covariance matrix are constructed. Conventional MDL-based decision tree is applied to do model clustering. The penalty scaling factor α is set to 1. 50 utterances are used as test set.

Wall Street Journal CSR corpus is used to train CD-DNN-HMM acoustic model. Training set (SI-284) contains 78 hours utterances recorded by 284 native American English speakers. The acoustic features, extracted by a 25ms hamming window, shifted every 10ms, consist of 38 MFCCs plus log energy. Three states, left-to-right HMM triphone models, each state with 16 Gaussians components, diagonal covariance distribution, are trained. The phone set is constructed by grouping TIMIT phonemes into 40 phonemes. The total number of "senones" after state-tying is 2,754.

Acoustic models are further trained by DNN with all training data (SI-284)[18]. Our SI-DNN model is a 6 layer network, consisting of 1 input layer, 4 hidden layers, each layer with 2K units, and 1 output layer, with the same number of senones output as in CD-GMM-HMM. The input of DNN is MFCCs, which contains 5 left frames, the current frame and 5 right frames (429 dimensions). Each dimension is normalized to zero mean and unit variance. Our DNN is initialized with the Deep Belief Network (DBN) pre-training procedure [19]. All weights and bias are then discriminatively tuned using about 100 epochs.

3.2. Experimental Results and Analysis

3.2.1. *Convergence of Phonetic Clustering*

Fig. 3 shows the convergence property of KL divergence based phonetic clustering. BDL's 1000 utterances are used for clustering. And another 50 utterances are used for testing. We measure the total KL divergence between each sample and the centroid of phonetic cluster it belongs to. Seen from the results of the training and test set, the phonetic clustering is converged after 30 training iterations. We also set different initialized centroid to study whether this KL divergence based phonetic clustering is sensitive to the initialization. And we found that the total KL divergence on the test set is differed within 1% with different centroid initializations which proves that the phonetic clustering is quite robust to the initialization mainly due to that the samples are well structured since each dimension of the DNN output posterior has its phonetic meaning.

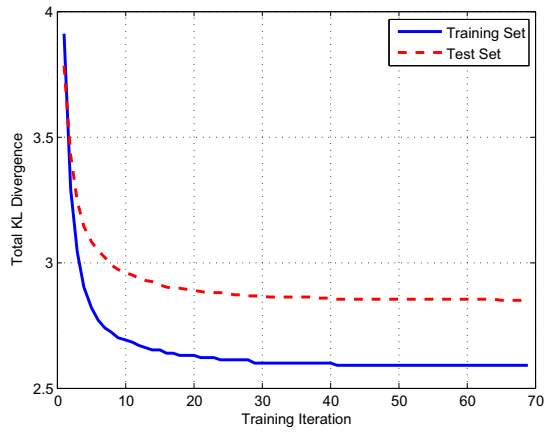

Figure 3: Convergence of Phonetic Clustering

3.2.2. *Senone Mapping vs. Phonetic Cluster Mapping*

Table 1 shows the log spectral distortion (LSD) between the converted speech and the target speaker's natural recordings of the same sentence. We compare 3 systems constructed with target speaker's 1000 utterances: TTS senone mapping(TSM), phonetic cluster mapping (PCM), and the TTS directly trained with target speaker's 1000 utterances which can be viewed as the upper bound of the performance. Both TSM and PCM approach achieve a performance close to the TTS upper bound. Even without transcriptions, PCM achieves a better result than TSM, due to that the phonetic clusters are constructed with the DNN output posterior distributions (phonetic representations) which is consistent with the DNN output posterior distributions based KLD matching process. On the other hand, TTS senones are constructed with acoustic features by maximizing the likelihood, which is not consistent with the KLD mapping process.

Table 1: LSD(dB) of TSM, PCM, TTS upperbound

Conversion Pair	TSM	PCM	TTS upperbound
SLT to BDL	4.68	4.56	4.10
CLB to SLT	4.51	4.21	4.06
RMS to BDL	4.61	4.47	4.10

3.2.3. *Amount of Data*

Table 2 shows the LSD on test set of speaker pair from SLT to BDL using the phonetic cluster mapping method. The number of utterances to construct target speaker's phonetic clusters is set from 10 to 1000. From the results, 100 utterances seems to be adequate to construct the target speaker's phonetic space. Even with only 30 seconds speech (10 utterances) of target speaker, the KLD-DNN voice conversion system can achieve fairly good intelligibility and decent similarity to the target speaker form 8 subjects' opinions after hearing testing stimuli.

Table 2: LSD(dB) on test set

# of utterances	10	50	100	300	500	1000
LSD(dB)	4.96	4.67	4.59	4.57	4.57	4.56

3.2.4. *Speaker Dependent DNN*

Fig. 4 shows the LSD on test set of speaker pair from SLT to BDL when we have source speaker and target speaker's pre-recorded speech with transcriptions to adapt the SI-DNN to SD-DNN. From the figure, when we use more data to adapt the SI-DNN to SD-DNN, the DNN output posteriors are more phonetically accurate. Thus the KL divergence based mapping can find more appropriate target speaker's TTS senones or phonetic clusters for the source speaker's speech frame.

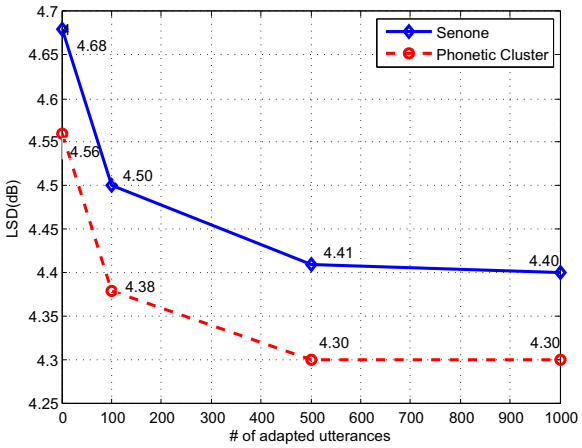

Figure 4: LSD(dB) with different # of adapted utterances

3.2.5. *KLD-DNN vs SEM-NN*

We compare KLD-DNN approach with the SEM-NN[5] based approach, our baseline, proposed previously. 3 systems are construsted. SEM-NN-100 is trained with 100 parallel utterances based on the sequence error minimization criterion. PCM-100 only uses target speaker's 100 untranscribed utterances to train a phonetic cluster mapping based voice conversion system. PCM-A-100 uses both source speaker and target speaker's 100 utterances to adapt the SI-DNN to get their own speaker specific DNN output posterior distributions. And then a phonetic cluster mapping based voice conversion system is trained. Our KLD-DNN based voice conversion outperform the conventional neural network based voice conversion significantly.

Table 3: LSD(dB) on test set

Conversion pair	PCM-100	PCM-A-100	SEM-NN-100
SLT to BDL	4.59	4.40	4.98
CLB to SLT	4.23	4.10	4.50
RMS to BDL	4.49	4.40	5.10

A subjective listening test for naturalness and similarity comparison of PCM-100 and SEM-NN-100 is also conducted with 8 subjects. 30 samples of 3 speaker conversion pairs are included in the test. PCM-100 achieves overwhelming preference than SEM-NN-100 with naturalness (96% vs. 4%), similarity (97% vs. 3%). Samples of the converted utterances are given on the web link: http://feng-long.github.io/VC

4. Conclusions

In this paper, we propose KLD-DNN based approach to voice conversion without parallel data between the source and target speakers. SI-DNN is used to equalize the speaker difference between source and target speaker in their DNN output posterior distribution phonetic space. KL divergence is used to map the source speaker's frame to the closest target speaker's TTS senone or phonetic cluster. Both the senone mapping and phonetic cluster mapping achieve performance close to the upperbound of TTS trained directly with target speaker's speech. Our KLD-DNN based approach significantly outperforms the conventional neural network based baseline both objectively and subjectively.

5. References

[1] Y. Stylianou, O. Cappe, and E. Moulines, "Continuous probabilistic transform for voice conversion," *IEEE Trans. on Audio Speech and Language Processing*, vol. 6, no. 2, pp. 131-142, 1998.

[2] T. Toda, A. Black, and K. Tokuda,"Voice conversion based on maximum-likelihood estimation of spectral parameter trajectory," *IEEE Trans. on Audio Speech and Language Processing*, vol. 15, no. 8, pp. 2222-2235, 2007.

[3] M. Narendranath, H. A. Murthy, S. Rajendran, and B. Yegnanarayana, "Transformation of formants for voice conversion using artificial neural networks," *Speech Communication.*, vol. 16, no. 2, pp. 207-216, 1995.

[4] S. Desai, A. Black, B. Yegnanarayana, K. Prahallad, "Spectral Mapping Using Artifical Neural Networks for Voice Conversion," *IEEE Trans. on Audio Speech and Language Processing*, vol. 18, no. 5, pp. 954-964, 2010.

[5] F-L. Xie, Y. Qian, Y. Fan, F. K. Soong, H. Li "Sequence Error(SE) Minimization Training of Neural Network for Voice Conversion," in *Proc. Interspeech*, pp. 2283-2287, 2014.

[6] L. Chen, Z. Ling, Y. Song, L. Dai, "Joint Spectral Distribution Modeling Using Restricted Boltzmann Machines for Voice Conversion," in *Proc. Interspeech*, pp. 3053-3056, 2013.

[7] Z. Wu, E. Chng, H. Li, "Conditional Restricted Boltzmann Machine For Voice Conversion," in *Proc. ChinaSIP*, pp. 104-108, 2013.

[8] H. Valbret, E. Moulines, and J. P. Tubach, "Voice transformation using PSOLA technique," in *Proc. ICASPP*, pp. 145-148, 1992

[9] M. Abe, S. Nakamura, K. Shikano, and H. Kuwabara, "Voice conversion through vector quantization," in *Proc. ICASPP*, pp. 655-658, 1988

[10] K. Tokuda, T. Yoshimura, T. Masuko, T. Kobayashi, T. Kitamura, "Speech Parameter Generation Algorithms For HMM-based Speech Synthesis," in *Proc. ICASSP*, pp. 1315-1318, 2000.

[11] S. Kullback, R. A. Leibler, "On information and sufficiency," *The Annals of Mathematical Statistics*, pp. 79–86, 1951

[12] J. Kominek and A. Black, "The CMU ARCTIC databases for speech synthesis," Tech. Rep. CMU-LTI-03-177, Language Technologies Institute, Carnegie Mellon University, 2003.

[13] D. Sundermann, A. Bonafonte, H. Ney, and H. Hoge,"A First Step Towards Text-Independent Voice Conversion," in *Proc. ICSLP*, 2004.

[14] H. Ye and S. J. Young, "Voice Conversion for Unkown Speakers," in *Proc. ICSLP*, 2004.

[15] D. Sundermann, H. Hoge, A. Bonafonte, H. Ney, A. Black, S. Narayanan, "Text-Independent Voice Conversion Based on Unit Selection," in *Proc. ICASSP*, pp. 81-84, 2006

[16] F-L. Xie, F. K. Soong, H. Li, "A KL Divergence And DNN Approach To Cross-Lingual TTS, in *Proc. ICASSP*, pp. 5515-5519, 2016.

[17] F. K. Soong and B. -H. Juang, "Line Spectrm Pair (LSP) and speech data compression," in *Proc. ICASSP*, pp. 1.10.1–1.10.4, 1984.

[18] W. Hu, Y. Qian, F. K. Soong, "A new DNN-based high quality pronunciation evaluation for computer-aided language learning(CALL)," in *Proc. Interspeech*, pp. 1886–1890, 2013.

[19] G. E. Hinton, S. Osindero, Y. W. Teh, "A fast learning algorithm for deep belief nets," in *Neural Computation*, vol. 18, no. 7, pp. 1527–1544, 2006

[20] Z. Wu, T. Virtanen, E. S. Chng, H. Li, "Exemplar-based sparse representation with residual compensation for voice conversion," *IEEE Trans. on Audio Speech and Language Processing*, vol. 22, no. 10, pp. 1506-1521, 2014.

[21] H. Ming, D. Huang, L. Xie, S. Zhang, M. Dong, H. Li, "Exemplar-based Sparse Representation of Timbre and Prosody for Voice Conversion," in *Proc. ICASSP*, pp. 5175–5179, 2016